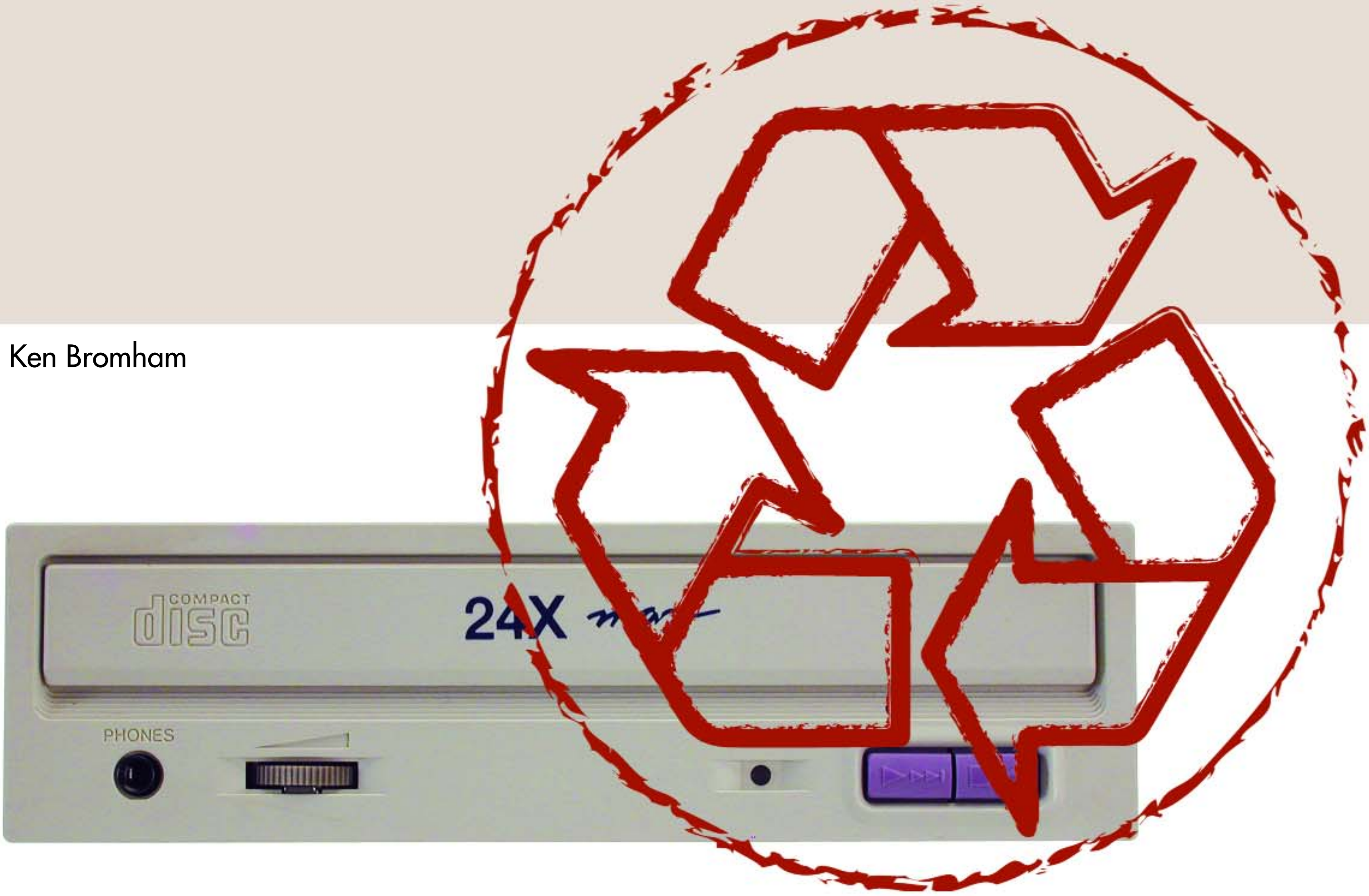
Ken Bromham



Pandora's Sound & Music Box

Recycle an old CD-ROM drive for (triggered) playback

This project was originally conceived for model making applications as a cheap way to provide high quality audio playback in response to a trigger signal (typically from a pushbutton or PIR detector), but we're pretty sure Elektor readers can find many other uses.

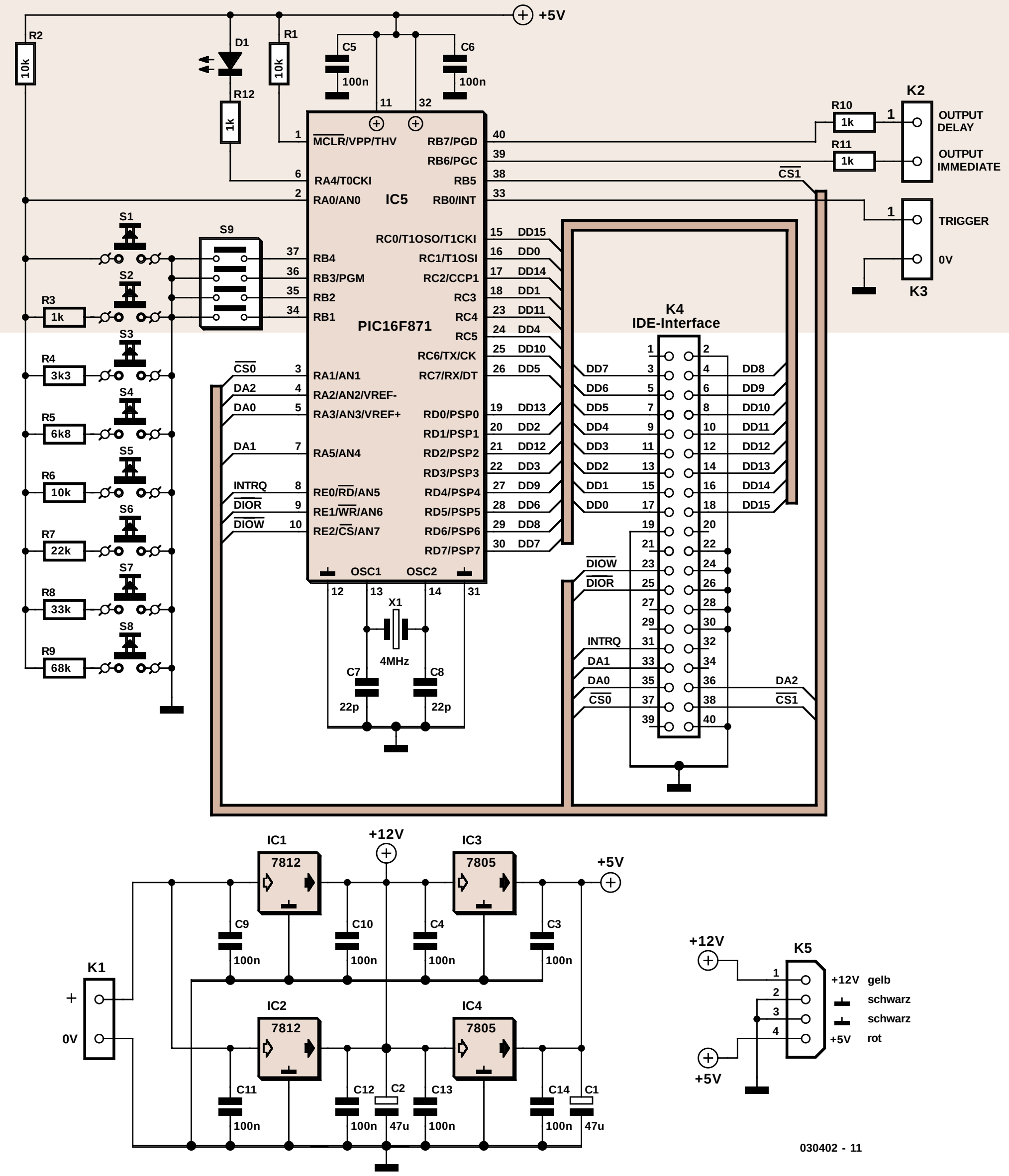


Figure 1. The modern version of Pandora's Box is a black one governing triggered response from an old CD-ROM drive playing back music or sound samples. You determine what 'horrors' (if any) lurk inside the box.

The standard low cost solution to creating auto-playback sounds is to use a record/playback chip (say, the ISD25XX series) but these devices are limited to about 3 kHz audio bandwidth, not to mention a lot of hiss and noise, so the sound quality is not the best unless you're specifically after for voice messages of the Stephen Hawking type.

The Elektor approach

Over the past few years we have received many requests from readers asking for a circuit that would allow them to use an old CD-ROM player for the sole purpose of playing back music CDs. In many cases, that is possible just by connecting a power supply and headphones to your drive, inserting a CD and pressing the Play button. Simple as this may sound, there are pitfalls, particularly with later CD-ROM drives that do not have a music playback button. Also, we really could not stand the thought of being unable to select and skip tracks, etc. In true Elektor fashion we wanted to be in control of things and make the old CD-ROM drive do something really useful. The alternative approach developed by Ken Bromham and described in this article makes use of a 40-pin PIC 16F87X microcontroller which, helped by a small number of external components, provides a versatile controller module that will interface to any old ATAPI CD-ROM drive. Mind you, ATAPI is not a brand, but a connectivity standard for 99% of all CD-ROM drives in PCs for home and office use. Don't worry about it, just start rummaging around in the attic or cellar, dig a CD drive out the IT skip at work, or commandeer the oldest (usually read-only) CD drive from the kiddies' PC leaving a note saying '*drive removed for scientific purposes*'. Never tell them you got the idea from Elektor, instead, tell them to keep using the CD/RW drive which is much faster, better, etc. or better still buy them a mini MP3 player and headphones.

COMPONENTS LIST

Resistors:
R1,R2,R6= 10kΩ
R3,R10,R11,R12 = 1kΩ
R4 = 3kΩ3
R5 = 6kΩ8
R7 = 22kΩ
R8 = 33kΩ
R9 = 68kΩ

Capacitors:
C1,C2 = 47µF 16V radial

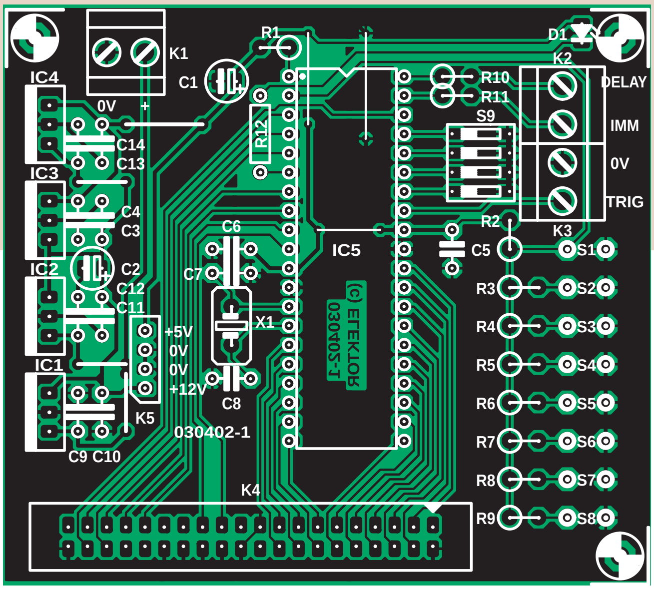


Figure 2. The printed circuit board is single-sided and contains seven wire links.

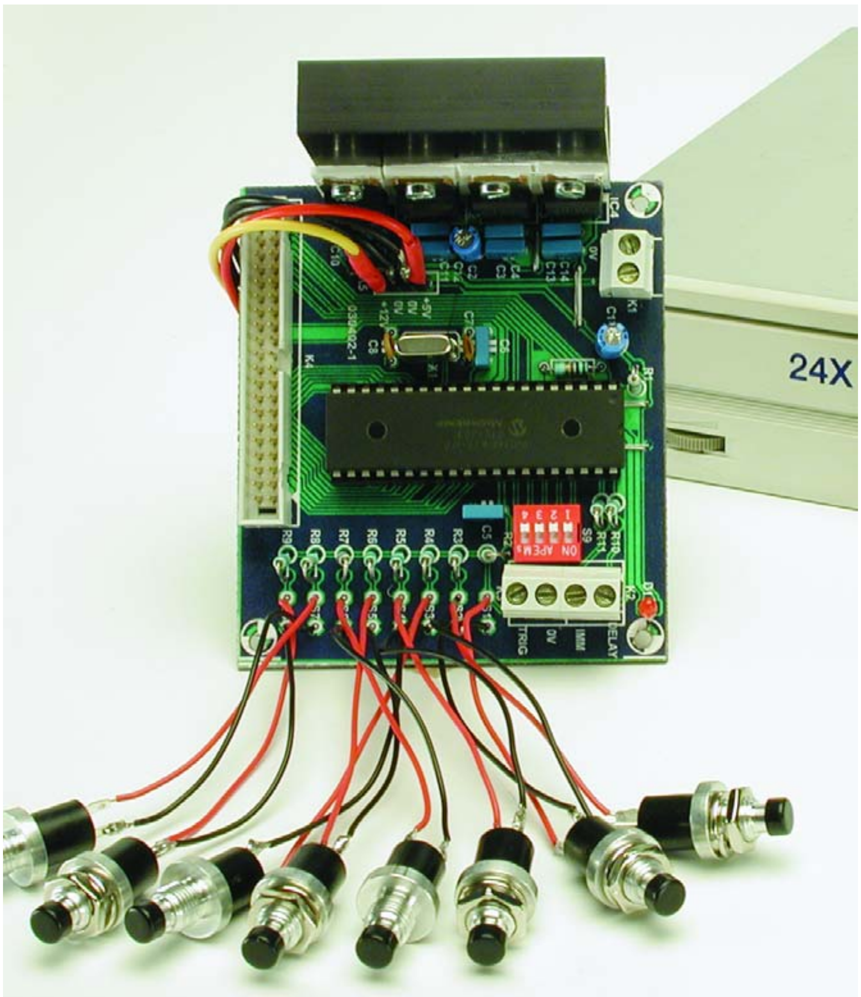


Figure 3. When in doubt about any constructional aspect, just use this photograph for guidance.

What can it do?

The controller module has two main modes of operation, single trigger use or multiple-trigger use, with further options selected by a 4-way DIL switch (psst… in fact both modes run simultaneously). There are also two digital switching outputs, asserted during playback, to allow other features (for example, lights or a motor) to be automatically switched on for the duration of the audio playback.

Single-trigger mode

The function here is very simple. Activating the single trigger input will cause playback of one track from the CD. There are four options (selected with a DIL switch), as follows :

DIL switch # 1
off = normally open contact for single trigger
on = normally closed contact for single trigger

DIL switch # 2
off = no response to trigger until end of track
on = respond anytime

DIL switch # 3
off = random track selection
on = sequential track selection

DIL switch # 4
off = keep disc continually spinning (defeat drive's inactivity timeout)
on = allow the CD-ROM drive to power down.

The latter options may require some elucidating. In the first case the playback will always start almost immediately as the disc is always spinning, but this may have some impact on the drive MTBF value (mean time between failure — you'll find it hard if not impossible to find data on this). In the second case, if the disc has stopped spinning, there will be a short 'spin up' delay before playback starts. No problem for applications that are likely to

C3..C6,C9-C14 = 100nF
C7,C8 = 22pF

Semiconductors:
D1=LED, red, low current
IC1,IC2 = 7812
IC3,IC4 = 7805
IC5 = PIC16F871/P, programmed, order code **030402-41** (see Readers Services page)

Miscellaneous:
S1-S8 = pushbutton, 1 make contact
S9 = 4-way DIP switch
X1 = 4.000MHz quartz crystal
K1,K2,K3 = 2-way PCB terminal block, lead pitch 5mm
K4 = 40-way boxheader
K5 = power supply plug for CD-ROM drive
PCB, order code **030402-1** (see Readers Services page)
Heatsink, e.g., Fisher SK59 (6 K/W)
Disk, PIC source and hex code files, order code **030402-11** or Free Download
CD-ROM drive

Free Downloads

PIC source and hex code files. File number: 030402-11.zip

PCB layout in PDF format. File number: 030402-1.zip

www.elektor-electronics.co.uk/dl/dl.htm, select month of publication.

spend a lot of time 'doing nothing'. There may be up to 24 tracks on the CD (the PIC chip is storing the table of contents data in its limited RAM). The multiple trigger inputs should be left open.

Multiple-trigger mode
In this mode you can have a maximum of eight separate triggers (for instance, push buttons). Pressing button 1 will always play track 1, button 2 will always play track 2, etc., up to and including track 8. It can be used with normally-open contacts only. If, for example, there are only three tracks on the CD then pressing buttons 4 – 8 will have no effect.

DIL switch 1 must be configured as normally open and the single trigger input left open.
DIL switch 2 is not relevant.
DIL switch 3 has the same function as above.
DIL switch 4 has the same function as above.

Digital switching outputs
Pandora's Sound & Music Box has two digital outputs for control of external devices like sounders, lamps, amplifiers, signal routers, door locks, you name it, anything can be controlled as long as it has a simple 0/5 V (TTL) digital control input, or can be switched on and off with a few mA of drive current.
The 'Output Immediate' output goes high immediately after triggering and remains high until end of playback.
The other output called 'Output Delay' goes high only after playback has started (that is, after any spin-up delay) and remains high until end of playback.

Circuit and construction

The circuit diagram shown in **Figure 1** has few surprises, basically showing a microcontroller sitting between a bunch of switches and some connectors. The heart of the circuit is a 40-pin PIC16F871 microcontroller which fortunately has enough input/output pins to connect to all of the necessary ATA interface lines with enough left over to handle the trigger inputs, option selections and the switching outputs. As can be seen from the schematic, only a handful of extra components are required. Note that the single trigger input RB0 (K3) and the option select inputs RB1-RB4 (S9) make use of internal, that is, invisible, pull-up resistors. All other port lines of the PIC16F871 are connected to the drive's IDE (ATAPI) interface via connector K4. Together with R2, resistors R3–R9 provide a simple potential divider network connected to input A0 on the PIC. The upshot is that a different voltage is applied to A0 depending on which button is pressed (multiple-trigger mode). This voltage is read by an internal A/D converter. It is assumed here that it is not necessary to distinguish multiple simultaneous button presses. If multiple trigger mode is not required then resistors R3–R9 and switches S2-S8 can be omitted, but R2 must be retained to keep input A0 pulled high.
The PIC ticks at 4 MHz as determined by quartz crystal X1 and its usual pair of small satellite capacitors, here identified as C7 and C8. The user-defined settings are read from DIL switch S9. One LED, D1, has been included to acts as a 'PIC awake' indicator (very useful!). Capacitors C5 and C6, finally, ensure the 5-V supply voltage to the PIC remains as clean as possible.
The circuit has been designed to operate from a single 15-18V DC supply, which should be 'heavy' enough to also supply the drive's 12-V line. Two paralleled 7805 fixed voltage regulators, IC3 and IC4, provide +5 V for the PIC and the CD-ROM drive's 5 V line. The 12-volt supply is realised in a similar way by two 7812s in parallel. Alternatively, an old PC power supply can be used to power the CD-ROM drive directly. In this case it is recommended to retain the 7805s and use 12 V from the pc power supply for the controller board. A heatsink will still be necessary.
Resistors R10 and R11 provide current limiting for the digital outputs and have been given the nominal value of 1 kΩ. The PIC chip can source/sink an absolute maximum of 25 mA for each pin, so the value of these resistors can be changed so long as this maximum is not exceeded. In any case, it is sufficient to drive a transistor/relay combination for example.
The PCB shown in **Figure 2** was designed for ease of use by you, the constructor. It is available ready-made through our Readers Services under number **030402-1**. Alternatively, you may decide to make your own board using the artwork file that can be downloaded free of charge from our website.
As there are only regular components to fit on the board we doubt the construction will present any problems. The simplest and cheapest component on the board, however, is often the one that's forgotten, causing major headaches and dozens of unnecessary emails of the 'Help it don't work' type! We're talking about the infamous **wire link**. There are seven of them on the board and they are best fitted before any other component so they're not forgotten. Bolt the voltage regulators onto a common heatsink (see **Figure 3** and the parts list), insulating washers are not required as all metal tabs are connected to ground. The PIC being the most expensive part, it deserves to be fitted into a 40-way DIL socket with good quality contacts.
Although the circuit diagram suggests that there are rather a lot of wires and other things to connect to the board, in reality the situation is not that bad as you can see from **Figure 4**. The cables

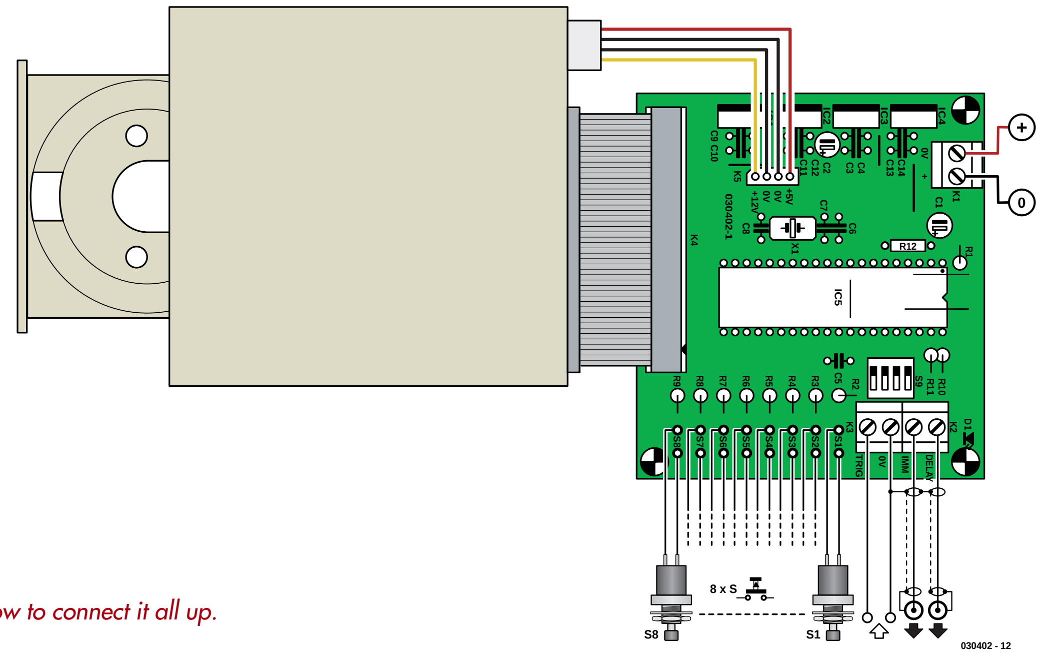


Figure 4. How to connect it all up.

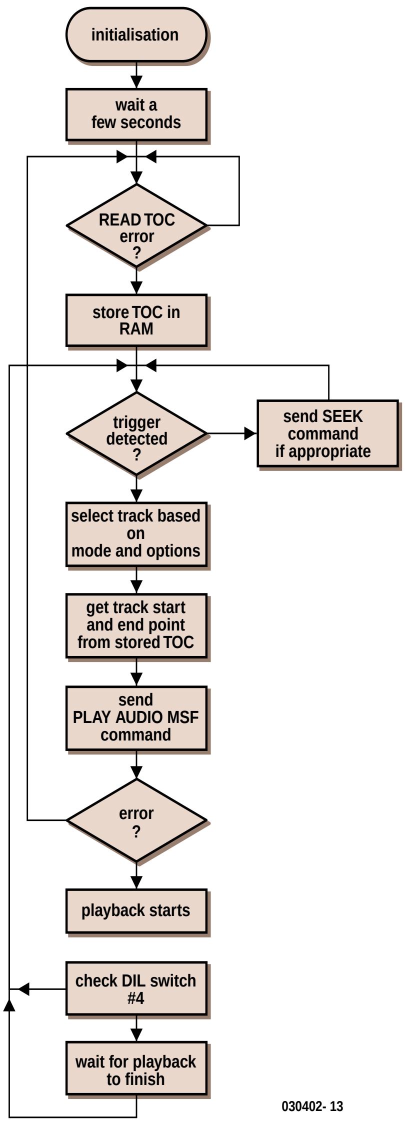


Figure 5. Use this flowchart of the main program if you don't fancy reading assembly code but still want to understand how the software works.

between the board and the CD-ROM drive, for example, are ready-made ones pulled from the junkbox or an old computer.

Program

The program that runs inside the PIC micro has been written in assembly language. The source code and hex file are available on floppy disk (anyone out there still using these?) or as free downloads from the Publishers' website. If you wish to program your own PIC, please feel free to do so using the files supplied. Alternatively a pre-programmed PIC is available from Readers Services, the order code being **030402-41**.

The source code supplied by Ken Bromham is well worth studying, even if you do not build the project. Ken succeeded in including plenty of comments so if you are familiar with this assembly language it should be possible to follow the program, despite some classic spaghetti code. If not, you may still want to grasp the 'broader lines' offered by the flowchart of the main program shown in **Figure 5**.

The actual ATAPI commands used by the PIC firmware are:

PLAY AUDIO MSF (play from specified start to end location. MSF = Minutes, Seconds, Frames, 75 Frames = 1 Second).

READ TOC (get the table of contents).

READ SUBCHANNEL (used to get the current audio status).

SEEK (position the head at start of track 1, but will also cause the disc to spin up, so used to defeat the CD-ROM drive inactivity timeout).

The READ SUBCHANNEL command is used at various points in the above sequence, whenever the program needs to know if playback is currently in progress or if playback has finished.

Testing

As usual, check for the presence of 5-V before inserting the PIC chip. The module can then be tested without connecting to a CD-ROM drive. Simply power up and check that the LED on pin A4 flashes a few times. Nothing else will happen, but this confirms that the PIC is up and running. Switch off, connect to CD-ROM drive, power up. The LED should flash a few times and then continue to flash until a disc is inserted, the tray is closed and the TOC (table of contents) successfully read. When the LED stops flashing, the module is ready to respond to a trigger and the different options can be experimented with.

Note that the CD-ROM drive must be configured as a MASTER device, and pin 1 on the PCB socket (K4) must go to pin 1 on the CD-ROM ATA interface socket (usually indicated by a red wire in the ribbon cable). When used with a single 12-V DC supply make sure this supply can provide a minimum of about 1.2 A. The audio output can be taken from the analogue out on the back of the CD-ROM drive or from the headphone jack a the front. To continue the low-cost theme, we recommend a cheap pair of active 'multimedia' speakers (whatever that means), unless, of course, you really want to build your own amplifier!

ATAPI protocol in brief

ATAPI (AT Attachment Packet Interface) devices use the same physical interface as ATA (AT Attachment) devices such as a hard disk drive, so it is necessary to understand how this works first. In summary, the ATA device has a limited number of 8-bit registers (for example,. COMMAND, STATUS) and a single 16 bit DATA register. The interface is of the parallel type with 16 bi-directional data lines, where only the lower 8 data lines are used for reading/writing the 8 bit registers. ATAPI devices use the same register set, although some registers have been renamed and serve a different function. Unfortunately this does not provide enough flexibility for the increased range of commands required, so the concept of a command packet was introduced along with a new ATA command, the 'ATAPI packet command'. To send a command to an ATAPI device the general procedure is to first write the (generic) ATAPI packet command to the device's COMMAND register, and then send the command packet by writing multiple times to the device's DATA register. For CD-ROM drives the command packet is 12 bytes in length and so 6 consecutive writes to the DATA register are required, sending 2 bytes each time. The command packet contains an opcode for the specific ATAPI command along with any additional parameters that are required.

Here is an example of the packet for the PLAY AUDIO MSF command:

Byte 0: Operation Code (0x47)
Byte 1: Reserved
Byte 2: Reserved
Byte 3: Start Location Minutes
Byte 4: Start Location Seconds
Byte 5: Start Location Frames
Byte 6: End Location Minutes
Byte 7: End Location Seconds
Byte 8: End Location Frames
Byte 9: Reserved
Byte 10: Reserved
Byte 11: Reserved

You can see that the command packet is padded with spare (reserved) bytes if necessary to give the 12-byte length.

Detailed documentation can be found on the web, www.t13.org is a good place to start looking, just be prepared for some serious bed-time reading.

ATA Interface pinning

Pin no.	Label	Description
1	$\overline{\text{HRESET}}$	Reset
2	GND	Ground
3	HD7	Data bus bit 7
4	HD8	Data bus bit 8
5	HD6	Data bus bit 6
6	HD9	Data bus bit 9
7	HD5	Data bus bit 5
8	HD10	Data bus bit 10
9	HD4	Data bus bit 4
10	HD11	Data bus bit 11
11	HD3	Data bus bit 3
12	HD12	Data bus bit 12
13	HD2	Data bus bit 2
14	HD13	Data bus bit 13
15	HD1	Data bus bit 1
16	HD14	Data bus bit 14
17	HD0	Data bus bit 0
18	HD15	Data bus bit 15
19	GND	GND
20	N/C	Key pin
21	DMARQ	DMA request
22	GND	Ground
23	$\overline{\text{HWR}}$	I/O write
24	GND	GND
25	$\overline{\text{HRD}}$	I/O read
26	GND	Ground
27	IORDY	I/O channel ready
28	SPSYNC:CSEL	Spindle sync or cable select
29	$\overline{\text{DMACK}}$	DMA acknowledge
30	GND	Ground
31	INTRQ	Interrupt request
32	$\overline{\text{IOCS16}}$	16 BIT I/O
33	HA1	Address bus bit 1
34	$\overline{\text{PDIAG}}$	Passed diagnostics
35	HA0	Address bus bit 0
36	HA2	Address bus bit 2
37	$\overline{\text{CS1FX}}$	Chip select 0
38	$\overline{\text{CS3FX}}$	Chip select 1
39	$\overline{\text{DASP}}$	Drive active/drive 1 present
40	GND	Ground

With compliments, Pandora

(030402-1)

There's only one limit to the applications of Pandora's Sound & Music Box: your imagination. Just couple the two notions 'some kind of trigger' to 'an audible, pre-recorded response' and away you go. Here are some possible applications to get you going, in fairly random order:

- **a voice message system employing the public address sound equipment in a large building. Very useful for guiding the public to fire exits in case of an emergency, when written notices are hardly ever seen, let alone read.**
- **an electronic dog barking in response to your doorbell; a track with More & Fiercer Dogs for really persistent callers that push the bell a second time (Pandora must have heard of Cerberus), gunfire for a third time, then police sirens, and so on.**
- **a door- or doormat-triggered muzak or 'welcome' generator.**
- **a voice guide in museums, triggered by visitors approaching an exhibit.**
- **a low-cost jingle-and-tune box for quizmasters and deejays.**
- **a language training aid.**
- **a spoken Callsign / 'CQ Contest' generator for radio amateurs.**

For all of the above applications, you will need to burn your own music CD. Programs to compile music or sound samples onto your own CDs abound in PC land, CoolEdit being one of the best known. Note however that you can't use MP3 files just like that — a suitable decoder will have to be added.